In the Claims:

15-19 Canceled

20. (Currently amended) A switch comprising:

a plurality of core modules;

a plurality of core controllers operating concurrently and independently, one core controller for each of said core modules;

a plurality of egress modules each having a link from each of said core modules; and

a plurality of ingress modules each having:

an ingress controller;

a plurality of ingress ports each having an ingress buffer; and

a link directed to each of said core modules;

wherein said ingress controller is operable to

sort packets arriving at said ingress buffer into ingress queues, each ingress queue corresponding to one of said egress modules;

issue packet-transfer requests each specifying an egress module;

distribute the packet-transfer requests among said plurality of core modules for scheduling so that each of said core controllers receives a portion of said transfer requests; and

~~The packet switch as claimed in claim 18 wherein the ingress controller~~

periodically determine[~~s~~] ingress-queue occupancy for each respective egress module and send[~~s~~] a capacity-request vector to a selected one of the core

controllers, the capacity-request vector including capacity-allocation requests for connections to a subset of the egress modules.

21. (Currently amended) The ~~packet~~ switch as claimed in claim 20 wherein each of said core modules computes a schedule in response to receiving a capacity-request vector, the schedule specifying, for each capacity request, time slots in a predefined time frame.

22. (Original) The switch as claimed in claim 21 wherein each core module comprises a single-stage space switch.

23. (Original) The switch as claimed in claim 22 wherein said space switch is a rotator space switch comprising an input rotator, a plurality of memory devices, and an output rotator.

24. (Original) The switch as claimed in claim 21 wherein the core modules, the ingress modules, and the egress modules are spatially distributed.

25. (Original) The switch as claimed in claim 24 wherein each core module comprises a number $S \geq 1$ of single-stage space switches, each space switch having a plurality of input ports each connecting to an ingress module and a plurality of output ports each connecting to an egress module.

26. (Original) The switch as claimed in claim 25 wherein said single-stage space switch is an optical switch.

27. (Original) The switch as claimed in claim 25 wherein each ingress module has a plurality of timing circuits each communicating with a time counter associated with a one of said core modules to realize time coordination between said each ingress module and said one of said core modules.

28. (Currently amended) The ~~packet~~ switch as claimed in claim 25 wherein each link carries a wavelength-division-multiplexed optical signal comprising S wavelength channels and wherein each of said single-stage space switches connects channels of the same wavelength.

29. (Original) A method of scheduling performed by a controller of a core module having at least one link from each of a plurality of ingress modules, said core module comprising a

plurality of S ≥1 space switches, each space switch having a plurality of input ports and a plurality of output ports, the method comprising steps of:

creating a data structure comprising

a first three-dimensional matrix having a space dimension *s* representing space switches associated with the core module, a space dimension *p* representing space-switch input ports, and a time dimension *t* representing time slots in a slotted time frame; and

a second three-dimensional matrix having said space dimension *s*, a space dimension π representing space-switch output ports, and said time dimension *t*;

receiving capacity-allocation requests from the ingress modules, each request specifying an input port *p*, an output port π, and a number K of time slots per time frame; and

selecting a space switch s and a time slot t and, if both entries {s, *p*, t} of the first matrix and {s,π, t} of the second matrix are free, allocating space-switch s and time-slot t and marking entries {s, *p*, t} and {s,π, t} as busy.

30. (Original) The method as claimed in claim 29 further including a step of repeating said selecting until at most K time slots are allocated.

31. (Original) The method as claimed in claim 29 further including a step of terminating a current connection by setting the value of K to zero.

32. (Original) The method as claimed in claim 30 wherein said selecting for each capacity-allocation request considers all time slots in said time frame, then all of said plurality of space switches.

33. (Original) The method as claimed in claim 32 further including a step of producing a scheduling matrix that associates each of said input ports with each of said output ports during each of said time slots.

34. (Original) The method as claimed in claim 33 wherein said all time slots are considered in a predetermined order.

35. (Original) The method as claimed in claim 33 wherein said all of said plurality of switches are considered in a predetermined order.

36. (Original) The method as claimed in claim 33 including the further steps of:

assembling said capacity-allocation requests into a capacity-request matrix each entry of which containing a requested capacity allocation; and

attempting to schedule each entry in the matrix to perform core reconfiguration.

37. (Original) The method as claimed in claim 36 including the further step of maintaining at each ingress module two scheduling matrices, one in current use and one in update mode.

38. (Original) The method as claimed in claim 37 wherein each time a core reconfiguration occurs, a scheduling matrix in use is swapped for a current scheduling matrix.

39. (Original) The method as claimed in claim 38 wherein an unused copy of the scheduling matrix is available for update after the core reconfiguration.

40. (Original) The method as claimed in claim 39 wherein rows in the scheduling matrix are executed sequentially, one per time slot, until a next core module reconfiguration occurs, and, after core module reconfiguration, processing continues at a next time slot.

41. (Currently amended) A distributed packet switch comprising:

a plurality of ~~m~~ channel cross connectors each having a plurality of ~~n~~ outer links and a plurality of ~~n~~ inner links;

a plurality of ~~n~~ core modules each of said core modules comprising a number of space switches;

a plurality of ~~m×n~~ edge modules each including an edge controller; and

a plurality of ~~n~~ core controllers, one core controller associated with each of said core modules;

wherein

each of said outer links connects to an edge module and includes a first number of ~~A~~ channels in each direction to and from ~~the~~ said edge module;

each of said inner links connects to a core module and includes a second number of ~~A~~ channels in each direction to and from ~~the~~ said core module.

said edge modules and said core modules are spatially distributed over a wide geographical area; and

each of said edge modules exchanges timing packets with each of said core modules to accomplish time coordination.

~~; and~~

~~said number of space switches per core module does not exceed the ratio A/n.~~

42. (Canceled)

43. (Canceled)

44. (Currently amended) The distributed packet switch as claimed in claim ~~43~~ 41 wherein each of said outer links is a wavelength-division-multiplexed link and each of said inner links is a wavelength-division-multiplexed link.

45. (Canceled)

46. (Currently amended) The distributed packet switch as claimed in claim ~~45~~ 41 wherein said means includes a timing circuit.

47. (Original) The distributed packet switch as claimed in claim 46 wherein each of said core modules has means for time coordination with each of said edge modules.

48. (Original) The distributed packet switch as claimed in claim 47 wherein said means includes a timing circuit.

49. (Original) The distributed packet switch as claimed in claim 48 wherein at least one of said space switches in each of said core modules operates in a time-division-multiplexed mode.

50. (Original) The distributed packet switch as claimed in claim 49 wherein at least two edge modules transmit time-division-multiplexed signals to said at least one of said space switches and adjust their transmission times so that the time-division-multiplexed signals are received at said at least one of said space switches in time alignment.

51. (Original) The distributed packet switch as claimed in claim 50 wherein one of said edge modules is collocated with a selected one of said core modules and hosts a controller that serves as the core controller of said one of said core modules.

52. (Original) The distributed packet switch as claimed in claim 50 wherein each of said edge modules is adapted to transmit capacity-allocation requests to any of said core modules.

53. (Original) The distributed packet switch as claimed in claim 52 wherein the core controller associated with said any of said core modules computes a schedule in response to receiving said capacity-allocation requests, the schedule specifying, for each capacity request, time slots in a predefined time frame.

54. (Original) The distributed packet switch as claimed in claim 53 wherein at least one of said space switches is an electronic space switch.

55. (Original) The distributed packet switch as claimed in claim 53 wherein at least one of said space switches is an optical space switch.

56. (Original) The distributed packet switch as claimed in claim 53 wherein at least one of said space switches is an electronic single-stage rotator switch.

57-66 Canceled

67. (Currently amended) The packet switch as claimed in claim ~~66~~ 72 wherein each of said core modules maintains its own time reference.

68. (Original) The packet switch as claimed in claim 67 wherein each of said ingress modules times the transmission of its data blocks to arrive at a selected one of said core modules at a time determined by said selected one of said core modules.

69-71 Canceled

72. (Currently amended) ~~The~~ A packet switch ~~as claimed in claim 66 wherein~~ comprising:

a plurality of egress modules, each for transmitting packets to subtending packet sinks;

a plurality of ingress modules each ingress module ~~has~~ having a plurality of ingress ports and an ingress controller, each of said ingress ports including:

an associated ingress buffer for receiving packets from subtending packet sources; and

means for sorting packets arriving in the ingress buffer into ingress queues each ingress queue corresponding to an egress module from which packets in said each queue are to egress from the switch for delivery to the subtending packet sinks,

a plurality of core modules each connecting to each ingress module and to each egress module and having its own controller for allocating and scheduling data paths of sufficient capacities to accommodate connection requests received from any ingress module to transfer packets to any egress module, said own controller operating independently of and concurrently with controllers of other core modules; and

at least one cross-connector connecting a subset of the ingress modules to the core modules,

wherein each ingress module includes means for routing a connection through at least one core module, and wherein said data paths carry data blocks of equal durations.

73. (Original) The packet switch as claimed in claim 72 wherein each ingress module is further adapted to send capacity-allocation requests to a one of said core modules.

74. (Original) The packet switch as claimed in claim 73 wherein each ingress module is further adapted to receive connection schedules from each of said core modules.

75. (Original) The packet switch as claimed in claim 73 wherein each of said connections is scheduled for transmission over time-slots of equal duration and each of said connection schedules includes time-slot identifiers for each scheduled connection.

76. (Original) The packet switch as claimed in claim 73 wherein each ingress module is further adapted to:

create a vector of pointers to the sorted packets; and

assemble said connection schedules into a scheduling matrix so that a non-blank entry in the scheduling matrix indicates an index of the vector of pointers.

77. (Original) The packet switch as claimed in claim 76 wherein said ingress controller receives capacity requirements from subtending sources and determines said capacity-allocation requests.

78. (Original) The packet switch as claimed in claim 76 wherein the ingress controller periodically determines a number of packets waiting in the ingress queues for each respective egress module and determines said capacity-allocation requests.

79. (Original) The switch as claimed in claim 75 wherein each core module comprises a plurality of single-stage rotator switches, each rotator switch having a number of input ports collectively adapted to accommodate a number of channels at least equal to the number of ingress modules and a number of output ports collectively adapted to accommodate a number of channels at least equal to the number of egress modules, each ingress module having at least one channel to each of the rotator switches, and each egress module having at least one channel from each of the rotator switches.

80. (Original) The packet switch as claimed in claim 79 wherein each ingress module is combined with an egress module to form an integrated edge module.

81. (Original) The switch as claimed in claim 75 wherein the core modules are co-located at one geographical location.

82. (Original) The switch as claimed in claim 75 wherein the core modules, the ingress modules, and the egress modules are spatially distributed.

83. (Original) The packet switch as claimed in claim 75 wherein each ingress module has a number of timing circuits at least equal to the number of core modules, each of the timing circuits being time-coordinated with a time counter associated with each of said core modules.

84. (Original) The switch as claimed in claim 83 wherein one edge module is co-located with each core module and said one edge module serves as a controller for the core module.

85. Canceled

86. (Currently amended) ~~The method as claimed in claim 85 including the further step of~~

A method of switching a packet through a switch comprising a plurality of ingress modules each having at least one ingress port, a plurality of egress modules each having at least one egress port, and a plurality of core modules, wherein each ingress module is coupled to each core module, each core module is coupled to each egress module, the method comprising the steps of:

one of said ingress modules receiving a packet from a subtending traffic source;

said one of said ingress modules sending a connection request to a selected one of the core modules, requesting a connection of a specified capacity to one of said egress modules;

said selected one of the core modules:

~~determining~~ available capacity;

updating said connection request according to said available capacity; and

returning the connection request to said one of the ingress modules; and

said one of the ingress modules sending the connection request to another of the core modules if said ~~feasible~~ available capacity ~~allocation~~ is less than said specified capacity.

87. (Original) The method as claimed in claim 86 wherein said any one of the core modules is selected at random.

88. (Original) The method as claimed in claim 86 wherein said any one of the core modules is selected according to a preferred order.

89. (Original) The method as claimed in claim 88 wherein said preferred order is specific to said one of the ingress modules and said one of the egress modules.

90-102 Canceled